7К

УДК 004.89:004.93

С.А. Ковтун, С.Н. Капитан, О.О. Савельев
Институт проблем искусственного интеллекта МОН Украины и НАН Украины
г. Донецк, Украина

О концепции создания интеллектуальных тестирующих систем

В статье рассматривается проблема построения интеллектуальной тестирующей системы. Проведено исследование и анализ методов, идей, технологий и алгоритмов, положенных в основу существующих систем тестирования с целью выявления уровня их интеллектуализации, а также разработана концепция интеллектуальной тестирующей системы.

Введение

Бурное развитие компьютерных и телекоммуникационных средств, а также создание всемирной сети Интернет, позволило использовать персональный компьютер не только как помощника в работе и научных вычислениях, но и как инструмент обучения и оценки знаний.

В настоящее время широкое распространение получили тестирующие системы, которые применяются в разнообразных областях и затрагивают различные сферы жизни общества. С их помощью проверяется уровень знаний, осуществляется подбор квалифицированных кадров, проводится сертификация персонала, производится анализ способностей человека и др. В данной статье был проведен анализ тестирующих систем с целью выявления уровня их интеллектуализации, а также разработана концепция создания интеллектуальной тестирующей системы.

Целью данной работы является исследование и анализ методов, идей, технологий и алгоритмов, положенных в основу существующих систем тестирования, а также разработка концепции интеллектуальной тестирующей системы.

Анализ тестирующих систем

Анализ существующих тестирующих систем, доступных через сеть Интернет с целью выявления уровня их интеллектуализации, осуществлялся по следующим критериям:

1) цели тестирования;
2) виды тестовых заданий;
3) выбор последовательности тестовых заданий;
4) критерии завершения тестирования;
5) способ информирования тестируемого;
6) вид представления результата теста;
7) характеристика предлагаемых вариантов ответов.

На основе проведенного анализа были получены следующие результаты:

– большинство тестирующих систем, существующих в настоящее время, основано на закрытой форме тестовых заданий с выбором одного варианта ответа из четырех предложенных, имеющего определенный весовой балл. Это свойство редко позволяет учесть правильные ответы на основополагающие вопросы по предложенной теме, а также неполные или частично правильные ответы;

– почти в каждой системе присутствует генератор тестовых заданий, который осуществляет их случайную выборку из общего числа имеющихся в базе. Иными словами последующее тестовое задание никак не учитывает правильность ответов, данных на предыдущие;
– во всех системах тестирование завершается лишь при прохождении всей строго определенной совокупности тестовых заданий. Это не позволяет сократить время тестирования за счет «идеальных» ответов на сложные тестовые задания, с помощью которых можно практически однозначно идентифицировать уровень знаний.

Можно сделать вывод, что практически ни в одной из доступных для анализа тестирующих систем (опубликованных и предлагаемых) не используются методы искусственного интеллекта.

В то же время применение интеллектуальных тестирующих систем позволяет:
– значительно улучшить качество тестирования;
– отслеживать успеваемость как отдельного пользователя, так и всей группы;
– давать советы для более эффективного усвоения материала;
– подробно обосновывать человеку, прошедшему тестирование, полученную им оценку;
– сократить время прохождения теста за счет завершения процесса тестирования после наступления значимого события;
– оказывать помощь преподавателю по совершенствованию заданий теста и занятий в целом [1].

Введение интеллектуальных алгоритмов

Наиболее простые тесты имеют фиксированное количество стандартных вопросов, неизменную систему оценки полученных ответов, статический алгоритм построения последовательности теста.

Следующим шагом развития таких систем стало увеличение базы данных вопросов и ответов, а как следствие – разнообразие задаваемых программой вопросов. Как правило, эти вопросы выбирались случайным образом из базы данных. Подобный алгоритм позволил создать иллюзию уникальности предлагаемых тестов. Случайный выбор вопроса давал возможность формировать тесты с лавинообразным нарастанием количества вариантов.

Параллельно идет усложнение и системы оценки ответа пользователя. Первым шагом является статическое изменение весовых коэффициентов правильности ответа на тот или иной вопрос. База данных разбивается на группы по уровню сложности. Вносятся изменения в систему управления тестом – она уже формирует тест, хотя и случайным образом, но так, чтобы вопросы были представлены в соответствии с уровнем сложности. Как правило, уровень сложности возрастает с номером вопроса. Прохождение теста уже допускается в произвольном порядке – к более ранним вопросам можно возвращаться. Это, вообще говоря, вносит в процесс тестирования определенную обучающую составляющую. Более поздние вопросы теста могут натолкнуть пользователя на правильный ответ на более ранний вопрос.

Второй шаг – динамическое изменение системы оценивания. Классический пример – предложение подсказки пользователю при неправильном ответе со снижением оценки за ответ.

Подсказок на один и тот же вопрос может быть и несколько, снижение баллов может быть пропорционально «ценности» подсказки. Пользователь может сам решать – использовать или не использовать ту или иную подсказку [2].

Традиционная компьютерная система контроля знаний предлагает каждому тестируемому ответить на одинаковое количество вопросов без учета качества ответов проверяемого на эти вопросы, после чего выполняется обработка результатов теста. Эти тесты, как правило, включают перечень вопросов с вариантами ответов, среди которых находится правильный. Доля правильных ответов определяет итоговую оценку уровня знаний тестируемого. Такие тесты можно назвать «жесткими», так как в них фактически используется двоичная система оценки правильности ответа на каждый вопрос (правильно или неправильно).

Основным недостатком «жестких» тестов является оценка неполных или неточных ответов как неправильных, т.е. приближенное, неполное знание ответа на вопрос квалифицируется как незнание ответа, а это далеко не всегда оправдано. В результате «жесткие» тесты недостаточно справедливо оценивают уровень знаний учащегося, занижают оценку. Поэтому выявляется необходимость работы с более «мягкими» тестами, позволяющими учитывать не только «знает» – «не знает», но и «частично знает».

Создание тестирующих систем, работающих с «мягкими» тестами, позволяет решать и некоторые вопросы сопровождающего обучения при тестировании. Так как они позволяют вскрывать «анатомию» подготовки правильного ответа, что можно использовать при разборе ответа обучаемого и поиске именно того момента, который вызвал затруднение в его рассуждениях [3].

Динамически может меняться и схема прохождения теста. То есть программа может задавать вопросы определенного уровня сложности до тех пор, пока не будет получен правильный ответ (несколько правильных ответов), не переходя к более сложным вопросам. Это требует большого числа вопросов, а следовательно – больших баз данных.

Еще одной реализацией интеллектуальных алгоритмов функционирования является возможность самоорганизации теста и его адаптации к уровню подготовки пользователей.

Имеется в виду следующее: при многократном использовании теста программа сама может перераспределять вопросы по уровню сложности, опираясь на данные, полученные при тестировании. Проще говоря, вопрос, на который по данным статистики было получено наименьшее число правильных ответов, автоматически переходит в разряд сложных, увеличивается его весовой коэффициент. Коэффициент вопроса, на который было получено наибольшее количество правильных ответов, напротив снижается.

Это дает возможность построения адаптивных тестов, которые самокорректируются под уровень пользователей. Две одинаковые в начале функционирования программы, применяясь в различных группах, через некоторое время будут существенно отличаться друг от друга.

В практической реализации этого принципа нет ничего сложного – программа просто ведет определенный файл протокола работы, из которого при запуске считываются данные. Однако, это – тоже один из интеллектуальных алгоритмов работы системы.

Интерактивность при работе с тестирующей программой традиционно сводилась лишь к некоторому разнообразию способов обмена информации между системой и пользователем, а также к различной степени эргономичности интерфейса. Под ограниченностью интерактивности понимается невозможность пользователя влиять на исходные условия поставленной задачи.

Идея допустить такую возможность вплотную приводит к созданию тестирующих программ с элементами искусственного интеллекта. В таких системах ответ пользователя влияет не только на тип следующего вопроса, но и на саму базу данных. Для этого локальная система управления тестирующей программы должна иметь дополнительную аналитическую подсистему [2].

Также существуют интеллектуальные алгоритмы оценивания:

На первоначальной стадии система выдвигает 4 гипотезы: «испытуемый знает материал на 5», «испытуемый знает материал на 4», «испытуемый знает материал на 3», «испытуемый не знает материал». В этом первоначальном состоянии система дает одинаковую вероятностную оценку всем данным гипотезам. После каждого тестового задания уровень доверия между гипотезами распределяется в зависимости от выбранного варианта ответа. При определенном ответе на тестовое задание система высчитывает и изменяет уровень доверия к каждой гипотезе. При достижении критической отметки какого-либо уровня доверия система информирует пользователя об его отметке, а также указывает на темы плохо или недостаточно усвоенные для достижения более высоких результатов [1].

Алгоритм выбора очередного тестового задания является предметом отдельного исследования. Существует достаточно большое количество способов выбора последовательности тестовых заданий, среди которых выделяют случайный порядок, минимаксный алгоритм, метод наискорейшего спуска и другие. Выбор того или иного способа зависит от сценария проведения теста. Комплект тестовых заданий может включать в себя сценарий, который содержит следующую информацию: описание алгоритма выбора последующего тестового задания; условие завершения тестирования; указания системе об информировании и записи результатов в базу данных.

Тесты как способ проверки знаний

К безусловным достоинствам тестирующих программ можно отнести простоту их реализации. Создать тестирующую программу, даже использующую описанные выше усложненные схемы, может даже начинающий программист.

Достоинством является также и удобство использования таких систем пользователем. Выбирать правильный ответ из предложенных легче, чем самостоятельно получить результат. На это требуется меньше сил и времени.

Достоинством (хотя и не бесспорным) является то, что сама идея тестирования, т.е. предложение нескольких вариантов ответов, косвенно стимулирует пользователя анализировать различные решения, а как следствие – более глубоко исследовать поставленную задачу.

Тесты достаточно привлекательны, так как существенно сокращают время ответа на вопросы, а проверку тестов позволяют вообще полностью автоматизировать. Это открывает огромные перспективы использования тестирующих программ в среде дистанционного обучения, т.к. позволяют полностью реализовать весь процесс тестирования без участия преподавателя. Результаты тестов могут сохраняться автоматически и использоваться в дальнейшем. Этот принцип используют системы автоматического управления учебным процессом, создающие индивидуальные базы данных на каждого пользователя.

Однако имеются и недостатки.

Основным недостатком является конечность баз данных. Для того, чтобы обеспечить успешное функционирование тестирующей программы в течение длительного времени, необходимо постоянно добавлять и расширять базы данных. Иначе постепенно программа перестанет выполнять свои функции.

Вторым недостатком является малая информативность ответа пользователя. Выбор правильного варианта ничего не говорит о том, как он был получен. Возможно, просто угадан.

Можно отнести к недостаткам тестирующих программ и то, что они накладывают определенные ограничения и на сложность предлагаемых задач. Тест проходится за один сеанс и не допускает возможности отложить ответ, подумать, вернуться к тому же вопросу спустя день или два. Таким образом, естественно, что в тест можно включать только задачи, не требующие относительно больших усилий при решении. Особенно это неудобно в задачах по высшей математике.

Либо задания должны быть слишком простыми, либо тест затягивается на колоссальное время, а пользователь вынужден, сидя перед компьютером, производить вычисления на бумаге. Это вообще девальвирует целесообразность применения компьютерного теста. Теряется всякое отличие от традиционной контрольной работы, а подробное письменное решение оказывается намного информативнее, чем выбранный окончательный ответ [2].

И все же представляется весьма перспективным применение тестирующих программ в составе интеллектуального обучающего комплекса.

Также следует отметить роль интеллектуальных алгоритмов в работе тестирующих программ. Применение интеллектуальных тестирующих систем позволяет отслеживать успеваемость как отдельного обучаемого, так и всей группы; давать советы для более эффективного усвоения материала; оказывать помощь преподавателю по совершенствованию заданий теста и занятий в целом.

Выводы

Предложенная концепция тестирующей системы дает возможность использовать результаты оценки интеллектуального уровня пользователя и предлагает набор тестов, адаптированный к уровню подготовки тестируемого. При этом система перераспределяет вопросы по уровню сложности, опираясь на данные, полученные при тестировании. Это дает возможность построения адаптивных тестов, которые самокорректируются под уровень пользователей. Применение интеллектуальных алгоритмов позволяет также динамически изменять систему оценивания и схему прохождения теста, что существенно повышает качество и скорость тестирования.

Интеллектуальные системы тестирования могут использоваться как отдельно для проверки уже накопленных знаний, так и в составе обучающего комплекса для более эффективного усвоения приобретенных.

Литература

1. Хохлова О.С. Интеллектуальная тестирующая система / О.С. Хохлова // Новые информационные технологии : тезисы докладов XVI Международной школы-семинара. – М. : МИЭМ, 2008. – 297 с.
2. Ларин А.А. О концепции построения тестирующих программ [Электронный ресурс] / А.А. Ларин. – Режим доступа : alexlarin.narod.ru/Stats/testprog.htm
3. Олейников Б.В. Создание тестирующее-обучающей системы в соответствии с международными стандартами IMS QTI / Б.В. Олейников, И.Н. Егоров // Материалы Третьей Всероссийской научно-практической конференции-выставки «Единая образовательная информационная среда: проблемы и пути развития».

С.А. Ковтун, С.Н. Капітан, О.О. Савєльєв

Про концепцію створення інтелектуальних тестувальних систем
У статті розглядається проблема побудови інтелектуальної тестувальної системи. Проведено дослідження та аналіз методів, ідей, технологій та алгоритмів, що покладені в основу наявних систем тестування з метою виявлення рівня їх інтелектуалізації, а також розроблена концепція інтелектуальної тестувальної системи.

Статья поступила в редакцию 28.08.2009.